

Discontinuous Galerkin Methods

Min-Hung Chen

Department of Mathematics, NCKU

2020 Summer Workshop for Scientific Computing
Sep. 1, 2020

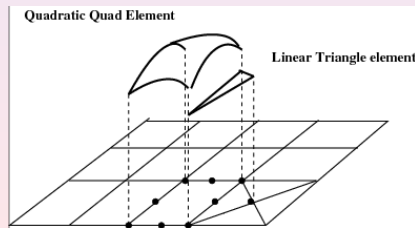
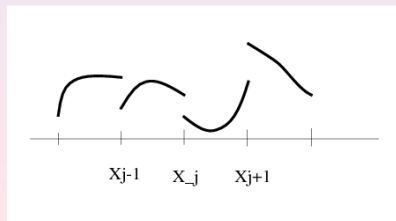
Outline

- 1 Introduction
 - DG for ODE
- 2 Linear Hyperbolic System
- 3 Steady-State Problems
 - Elliptic Problems
 - Fenics and Colab
- 4 HDG
 - Elliptic Model Problem
 - Matlab tools for HDG - HDG3D
- 5 Summary

Introduction

What is the Discontinuous Galerkin Method?:

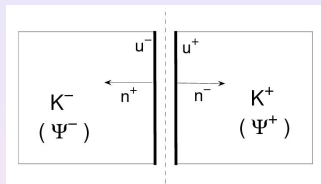
- a method between a finite element and a finite volume method
- local to the generating element
- a practical framework for the development of high-order accurate methods using unstructured grids



A short (and biased) historical overview of the DGM¹

- First DG method introduced in 1973 by Reed and Hill for linear transport.
- First studied in 1974 by Lesaint and Raviart.
- Extended to nonlinear hyperbolic conservation laws in the 90's by B. Cockburn and C.-W. Shu.
- Extended to compressible flow in 1997 first by F. Bassi and S. Rebay.
- New DG methods for diffusion appear and some old ones (the IP methods of the late 70's) are resuscitated. A unified analysis is proposed in 2002 by D. Arnold, F. Brezzi, B. Cockburn and D. Marini.
- They clash with the well-established mixed and continuous Galerkin methods. In response, the HDG methods are introduced in 2009 by B. Cockburn, J. Gopalakrishnan and R. Lazarov. The HDG methods are strongly related to the hybrid methods and to the hybridization techniques of the mid 60's introduced as implementation techniques for mixed methods.

¹<https://www.lacan.upc.edu/dg2017/>



Jump Operator:

$$\begin{aligned} \llbracket w \rrbracket &= \begin{cases} w|_{K^+} \mathbf{n}^+ + w|_{K^-} \mathbf{n}^- & \text{on } \partial K^+ \cap \partial K^- \\ w \mathbf{n} & \text{on } \partial \Omega \end{cases} \\ \llbracket \mathbf{v} \rrbracket &= \begin{cases} \mathbf{v}|_{K^+} \cdot \mathbf{n}^+ + \mathbf{v}|_{K^-} \cdot \mathbf{n}^- & \text{on } \partial K^+ \cap \partial K^- \\ \mathbf{v} \cdot \mathbf{n} & \text{on } \partial \Omega \end{cases} \end{aligned}$$

Average Operator:

$$\begin{cases} \{w\} = \begin{cases} \frac{1}{2}(w|_{K^+} + w|_{K^-}) & \text{on } \partial K^+ \cap \partial K^- \\ w & \text{on } \partial\Omega \end{cases} \\ \{v\} = \begin{cases} \frac{1}{2}(v|_{K^+} + v|_{K^-}) & \text{on } \partial K^+ \cap \partial K^- \\ v & \text{on } \partial\Omega \end{cases} \end{cases}$$

ODE²

- Equation

$$\frac{d}{dt}u(t) = f(t), \quad t \in (0, T), \quad u(0) = u_0.$$

- Partition of $(0, T) = \{t^n\}_{n=0}^N$, $I^n = (t^n, t^{n+1})$:

- Local Basis: $\{\phi_j^n\}_{j=0}^k$; (for example, $\phi_j^n(t) = P_j(\frac{2t-t^{n+1}-t^n}{t^{n+1}-t^n})$)

- weak form:

$$\int_{I^n} \frac{d}{dt}u_h(s)v(s) ds = \int_{I^n} f(s)v(s) ds,$$

or

$$-\int_{I^n} u_h(s) \frac{d}{dt}v(s) ds + u_h v|_{t^n}^{t^{n+1}} = \int_{I^n} f(s)v(s) ds,$$

- DG form:

$$-\int_{I^n} u_h(s) \frac{d}{dt}v(s) ds + \widehat{u_h} v|_{t^n}^{t^{n+1}} = \int_{I^n} f(s)v(s) ds,$$

²B. Cockburn, Discontinuous Galerkin methods, ZAMM Z. Angew. Math. Mech. 83

Numerical Flux

- DG form: $-\int_{I^n} u_h(s) \frac{d}{dt} v(s) ds + \widehat{u_h} v|_{t^n}^{t^{n+1}} = \int_{I^n} f(s) v(s) ds,$
- Upwinding Numerical Flux $\widehat{u_h}(t^n) = \begin{cases} u_0, & \text{if } t^n = 0, \\ u_h(t^{n-}), & \text{otherwise.} \end{cases}$
- Assuming Numerical Solution: $u_h|_{I^n}(t) = \sum_{i=0}^k a_i^n \phi_i^n(t)$
- Numerical Scheme: On I^n , for $j = 0, \dots, k$,

$$\begin{aligned}
 & -\int_{I^n} \sum_{i=0}^k a_i^n \phi_i^n(s) \frac{d}{dt} \phi_j^n(s) ds + \sum_{i=0}^k a_i^n \phi_i^n \phi_j^n(t^{n+1}) - \sum_{i=0}^k a_i^{n-1} \phi_i^{n-1} \phi_j^n(t^n) \\
 & = \int_{I^n} f(s) \phi_j^n(s) ds.
 \end{aligned}$$

- Matrix Equation: $M \begin{bmatrix} a_0^n \\ \vdots \\ a_k^n \end{bmatrix} = F$

Properties of the DG methods

- Discontinuous from element to element.
- Locally conservative

$$\widehat{u_h}|_{t_n}^{t^{n+1}} = \int_{I^n} f(s) ds,$$

- Relation between the residuals and jumps

$$-\int_{I^n} u_h(s) \frac{d}{dt} v(s) ds + \widehat{u_h} v|_{t_n}^{t^{n+1}} = \int_{I^n} f(s) v(s) ds,$$

Integration by parts,

$$\int_{I^n} \frac{d}{dt} u_h(s) v(s) ds + (\widehat{u_h} - u_h) v|_{t_n}^{t^{n+1}} = \int_{I^n} f(s) v(s) ds,$$

Or,

$$\int_{I^n} R(s) v(s) ds = (u_h - \widehat{u_h}) v|_{t_n}^{t^{n+1}}$$

Take $v = 1$,

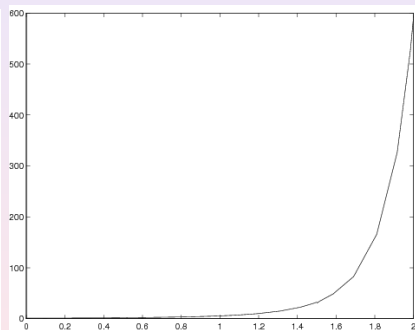
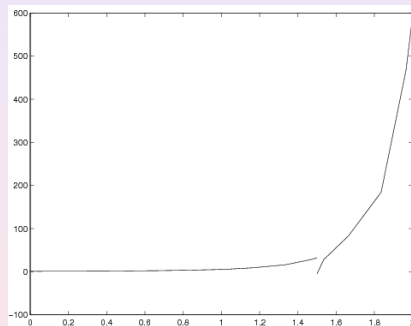
$$\int_{I^n} R(s) ds = \llbracket u_h \rrbracket(t^n)$$

Example

- $$\frac{d}{dt}u = e^t u(t), \quad t \in (0, 2), \quad u(0) = 1.$$
- dg_ode.zip
- Rate of Convergence and super-convergence (P^3)

N	L^2 -error	order	L^∞ node	order
2	0.50E+02	-	0.45E+02	-
4	0.66E+01	2.92	0.11E+01	5.33
8	0.63E-00	3.40	0.14E-01	6.34
16	0.44E-01	3.83	0.11E-03	6.93
32	0.28E-02	4.00	0.87E-06	7.01
64	0.17E-03	4.03	0.68E-09	7.02

Numerical Solutions of P^3 (Left) and P^5 (Right)



● Relation between the residuals and jumps

$$\int_{t^n}^{t^{n+1}} R(s) ds = \llbracket u_h \rrbracket(t^n)$$

	Interval 1			Interval 2		
k	Error	$\llbracket u \rrbracket$	Ratio	Error	$\llbracket u \rrbracket$	Ratio
1	0.33E-01	0.19E+00	0.17	0.19E-00	0.11E+01	0.17
2	0.35E-02	0.26E-01	0.13	0.27E-01	0.20E-00	0.14
3	0.34E-03	0.31E-02	0.11	0.34E-02	0.30E-01	0.12
4	0.32E-04	0.33E-03	0.097	0.41E-03	0.40E-02	0.10
5	0.28E-05	0.32E-04	0.087	0.45E-04	0.50E-03	0.090
6	0.24E-06	0.30E-05	0.080	0.47E-05	0.57E-04	0.082
	Interval 3			Interval 4		
k	Error	$\llbracket u \rrbracket$	Ratio	Error	$\llbracket u \rrbracket$	Ratio
1	0.19E-00	0.11E+02	0.17	0.73E+02	0.96E+02	0.76
2	0.43E-01	0.30E+01	0.14	0.29E+02	0.17E+03	0.18
3	0.72E-02	0.60E-00	0.12	0.46E+01	0.37E+02	0.12
4	0.11E-03	0.11E-01	0.10	0.95E-00	0.87E+01	0.11
5	0.16E-04	0.17E-02	0.094	0.18E-00	0.18E+01	0.098
6	0.21E-05	0.25E-03	0.085	0.32E-01	0.36E-00	0.089

Advection Equation

- Equation ($\Omega \subset \mathbb{R}^d$)

$$\begin{aligned} u_t + \nabla \cdot (\mathbf{a}u) &= 0 \text{ in } \Omega \times (0, T), \\ u(t=0) &= u_0, \text{ on } \Omega. \end{aligned}$$

- Triangulation: \mathcal{T}_h
- Local Space: $V(K)$ for $K \in \mathcal{T}_h$
- weak form: For all $v \in V(K)$

$$\int_K (u_h)_t v \, dx + \int_K \nabla \cdot (\mathbf{a}u_h) v \, dA = 0,$$

or

$$\int_K (u_h)_t v \, dx - \int_K \mathbf{a}u_h \cdot \nabla v \, dx + \int_{\partial K} \mathbf{a}u_h \cdot \mathbf{n}v \, ds = 0,$$

- DG form:

$$\int_K (u_h)_t v \, dx - \int_K \mathbf{a}u_h \cdot \nabla v \, dx + \int_{\partial K} \widehat{\mathbf{a}u_h} \cdot \mathbf{n}v \, ds = 0,$$

Numerical Flux and Stability

$$u_t + \nabla \cdot (\mathbf{a}u) = 0,$$

- Stability for the transport equation

$$\frac{1}{2} \int_{\mathbb{R}^d} u^2(x, T) dx + \frac{1}{2} \int_0^T \int_{\mathbb{R}^d} \nabla \cdot \mathbf{a}(x) u^2(x, t) dx dt = \frac{1}{2} \int_{\mathbb{R}^d} u_0^2(x) dx$$

Stable if $\nabla \cdot \mathbf{a} \geq 0$

Linear scalar equation

- DG form

$$\int_K (u_h)_t v \, dx - \int_K \mathbf{a} u_h \cdot \nabla v \, dx + \int_{\partial K} \widehat{\mathbf{a} u_h} \cdot \mathbf{n} v \, ds = 0,$$

- Stability for the DG methods (Take $v = u_h$)

$$\frac{1}{2} \int_{\mathbb{R}^d} u_h^2(x, T) \, dx + \frac{1}{2} \int_0^T \int_{\mathbb{R}^d} \nabla \cdot \mathbf{a}(x) u_h^2(x, t) \, dx \, dt + \int_0^T \Theta_h(t) \, dt = \frac{1}{2} \int_{\mathbb{R}^d} u_{h,0}^2(x) \, dx,$$

where

$$\begin{aligned} \Theta_h(t) &= \sum_{K \in \mathcal{T}_h} \left(\frac{1}{2} \int_K \nabla \cdot (\mathbf{a}(x) u_h^2)(x, t) \, dx + \int_{\partial K} \widehat{\mathbf{a} u_h}(x, t) \cdot \mathbf{n} u_h(x, t) \, ds \right) \\ &= \sum_{K \in \mathcal{T}_h} \int_{\partial K} (\widehat{\mathbf{a} u_h} \cdot \mathbf{n} u_h(x, t) - \frac{1}{2} \mathbf{a} u_h^2 \cdot \mathbf{n}) \, ds \\ &= \sum_{e \in \mathcal{E}_h} \int_e (\widehat{\mathbf{a} u_h} - \mathbf{a}\{u_h\}) \cdot \llbracket u_h \rrbracket \, ds. \end{aligned}$$

- Numerical Flux: ($\Theta_h(t) \geq 0$)

- ▶ General: $\widehat{\mathbf{a} u_h}(t^n) = \mathbf{a}\{u_h\} + C \llbracket u_h \rrbracket$
- ▶ Up-winding: $C = \frac{1}{2} |\mathbf{a} \cdot \mathbf{n}| Id \Rightarrow \widehat{\mathbf{a} u_h}(t^n) = \mathbf{a} \lim_{\epsilon \downarrow 0} u_h(x - \epsilon \mathbf{a})$
- ▶ Lax-Friedrichs: $C = \frac{1}{2} |\mathbf{a}| Id \Rightarrow \widehat{\mathbf{a} u_h}(t^n) = \mathbf{a}\{u_h\} + \frac{1}{2} |\mathbf{a}| \llbracket u_h \rrbracket$

DG Scheme

- DG form

$$\int_K (u_h)_t v \, dx - \int_K \mathbf{a} u_h \cdot \nabla v \, dx + \int_{\partial K} \widehat{\mathbf{a} u_h} \cdot \mathbf{n} v \, ds = 0,$$

$$\widehat{\mathbf{a} u_h}(t^n) = \mathbf{a}\{u_h\} + C \llbracket u_h \rrbracket$$

- Assuming Numerical Solution: $u_h|_K(t, x) = a_i(t)\phi_i(x)$
- Numerical Scheme: On K , for $j = 0, \dots, k$,

$$\int_K (a_i)_t \phi_i(x) \phi_j(x) \, dx - \int_K \mathbf{a} a_i \phi_i(x) \cdot \nabla \phi_j(x) \, dx + \int_{\partial K} \widehat{\mathbf{a} u_h} \cdot \mathbf{n} \phi_j(x) \, ds = 0,$$

- ODE

$$M \frac{d}{dt} U = N U$$

Example

$$u_t + u_x = 0 \quad \text{in} \quad (0, T = 1) \times (0, 1)$$

$$u(t = 0) = \sin 2\pi x \quad \text{on} \quad (0, 1)$$

Rate of Convergence (P^4 with RK 5)

N	L^2 -error	order
10	0.17E-05	-
20	0.52E-07	5.01
40	0.16E-08	5.01
80	0.51E-10	5.00
160	0.16E-11	5.00

Procedures to implement the RK-DG methods

- Conservation Form:

$$u_t + \nabla \cdot \mathbf{f}(u) = 0,$$

- DG space discretization

$$\int_K (u_h)_t v_h dx - \int_K \mathbf{f}(u_h) \cdot \nabla v_h dx + \int_{\partial K} \widehat{\mathbf{f}(u_h)} \cdot \mathbf{n}_K v_h ds = 0.$$

Here, the proper definition of $\widehat{\mathbf{f}(u_h)}$ is essential for the stability and convergence of the method.

- RK time discretization:

$$\frac{d}{dt} u_h = L(u_h)$$

Wave Equation

- Equation

$$u_{tt} - c^2 \Delta u = 0 \text{ in } \mathbb{R}^d \times (0, T).$$

- First-order system

$$U_t + \nabla \cdot F(U) = 0, \text{ in } \mathbb{R}^d \times (0, T),$$

where

$$U = \begin{pmatrix} q_1 \\ \vdots \\ q_d \\ u \end{pmatrix}, \quad F(U) = -c \begin{pmatrix} u & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & u \\ q_1 & \dots & q_d \end{pmatrix}.$$

- Triangulation: \mathcal{T}_h
- Local Space: $U(K) = P^k(K) \times \dots \times P^k(K)$ for $K \in \mathcal{T}_h$
- DG form: For all $V \in U(K)$

$$\int_K (U_{hi})_t V_i - \int_K F_{ij}(U_h) V_{i,j} + \int_{\partial K} \widehat{F}_{ij} \mathbf{n}_j V_i \, dx = 0,$$

Wave Equation

- Equation

$$u_{tt} - c^2 \Delta u = 0 \text{ in } \mathbb{R}^d \times (0, T).$$

- First-order system

$$U_t + \nabla \cdot F(U) = 0, \text{ in } \mathbb{R}^d \times (0, T),$$

where

$$U = \begin{pmatrix} q_1 \\ \vdots \\ q_d \\ u \end{pmatrix}, \quad F(U) = -c \begin{pmatrix} u & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & u \\ q_1 & \dots & q_d \end{pmatrix}.$$

- Triangulation: \mathcal{T}_h
- Local Space: $U(K) = P^k(K) \times \dots \times P^k(K)$ for $K \in \mathcal{T}_h$
- DG form: For all $V \in U(K)$

$$\int_K (U_{hi})_t V_i - \int_K F_{ij}(U_h) V_{i,j} + \int_{\partial K} \widehat{F}_{ij} \mathbf{n}_j V_i \, dx = 0,$$

Numerical Flux and Stability

- DG form: for all $V \in U(K)$

$$\int_K (U_{hi})_t V_i - \int_K F_{ij}(U_h) V_{i,j} + \int_{\partial K} \widehat{F}_{ij} \mathbf{n}_j V_i \, dx = 0,$$

- Stability for the DG methods (Take $V = U_h$)

$$\frac{1}{2} \int_{\mathbb{R}^d} U_h^2(x, T) \, dx + \frac{1}{2} \int_0^T \Theta_h(t) \, dt = \frac{1}{2} \int_{\mathbb{R}^d} U_h^2(x, 0) \, dx,$$

where

$$\Theta_h(t) = \sum_{e \in \mathcal{E}_h} \int_e (\widehat{F}_{ij} - \{F_{ij}\}) \cdot \llbracket U_h, i \rrbracket \, dx.$$

- Numerical Flux

$$\widehat{F}_{ij} = \{F_{ij}\} + C_{ijkl} \llbracket U_{hk} \rrbracket_l$$

Examples of the DG methods

Numerical Flux

$$\widehat{F}_{ij} = \{F_{ij}\} + C_{ijkl} \llbracket U_{hk} \rrbracket_l$$

- Up-winding:

$$\widehat{F}_{ij} = \{F_{ij}\} + \frac{|c|}{2} \llbracket q \rrbracket \delta_{ij} + \frac{|c|}{2} \llbracket u \rrbracket_j \delta_{i(d+1)}$$

- Lax-Friedrichs:

$$\widehat{F}_{ij} = \{F_{ij}\} + \frac{|c|}{2} \llbracket q_i \rrbracket_j \delta_{ij} + \frac{|c|}{2} \llbracket u \rrbracket_j \delta_{i(d+1)}$$

- Generalization of the Up-winding flux :

$$\widehat{F}_{ij} = \{F_{ij}\} + (C_{22} \llbracket q \rrbracket - C_{12} \cdot \llbracket u \rrbracket) \delta_{ij} + (C_{12} \llbracket q \rrbracket + C_{11} \cdot \llbracket u \rrbracket) \delta_{ij}$$

RK-DG Scheme

- DG form

$$\int_K (U_{hi})_t V_i - \int_K F_{ij}(U_h) V_{i,j} + \int_{\partial K} \widehat{F}_{ij} \mathbf{n}_j V_i dx = 0,$$

$$\widehat{F}_{ij} = \{F_{ij}\} + C_{ijkl} \llbracket U_{hk} \rrbracket_l$$

- Numerical Solution: $U_h|_K(t, x) = a_i(t) \phi_i(x)$
- Numerical Scheme: On K , for $j = 0, \dots, k$,

$$\int_K (a_i)_t \phi_i(x) \cdot \phi_j(x) dx - \int_K F(a_i \phi_i(x)) \cdot \nabla \phi_j(x) dx + \int_{\partial K} \widehat{F} \cdot \mathbf{n} \phi_j(x) ds = 0,$$

- ODE

$$M \frac{d}{dt} U = NU$$

Numerical Experiments

1D Test: Wave equation with sound speed $c(x) = \frac{1}{\sqrt{\mu\epsilon}}$

$$\partial_t u - c(x)^2 \partial_x v = 0, \quad \text{in } (0, T) \times (-2, 2)$$

$$\partial_t v - \partial_x u = 0, \quad \text{in } (0, T) \times (-2, 2)$$

$$c(x) = \begin{cases} 2 & \text{if } x \in (-1, 1) \\ 1 & \text{otherwise} \end{cases}$$

Initial conditions:

$$u(x, t = 0) = \phi(x), \quad x \in (-2, 2)$$

$$v(x, t = 0) = -\phi(x), \quad x \in (-2, 2)$$

Boundary Condition: (Transparent BC)

$$u + v = 0 \quad \text{at } x = 2$$

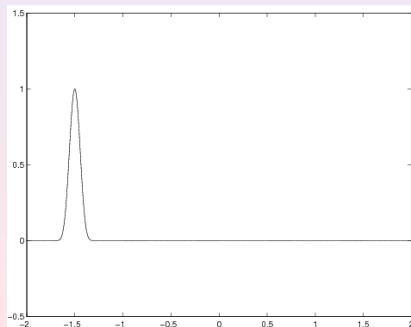
$$u - v = 2\phi(x - t) \quad \text{at } x = -2$$

(Chen, Cockburn, Reitich, J. Sci. Comp. 2005.)

Profile

$$\phi(x) = \psi\left(\frac{x + 1 + 0.5}{0.5}\right),$$

$$\psi(y) = \begin{cases} (2y - 1)^{10}(2y + 1)^{10} & \text{if } |y| < .5, \\ 0 & \text{otherwise.} \end{cases}$$


 $\phi(x - 1)$

Rate of Convergence

h	L^∞ -error	order	L^2 -error	order	L^1 -error	order
0.1250	0.64E-01	0.00	0.13E-01	0.00	0.25E-01	0.00
0.0625	0.17E-02	5.25	0.34E-03	5.28	0.63E-03	5.32
0.0312	0.60E-04	4.81	0.52E-05	6.01	0.86E-05	6.19
0.0156	0.40E-05	3.90	0.26E-06	4.31	0.34E-06	4.68
0.0078	0.25E-06	3.97	0.16E-07	4.00	0.21E-07	4.02
0.0039	0.16E-07	3.99	0.10E-08	4.00	0.13E-08	4.00

P^3 -elements and SSP-RK4 scheme at $T = 1.5$

h	L^∞ -error	order	L^2 -error	order	L^1 -error	order
0.1250	0.92E-02	0.00	0.19E-02	0.00	0.38E-02	0.00
0.0625	0.12E-03	6.26	0.12E-04	7.40	0.23E-04	7.40
0.0312	0.37E-05	5.04	0.22E-06	5.70	0.26E-06	6.43
0.0156	0.12E-06	4.97	0.70E-08	5.00	0.76E-08	5.11
0.0078	0.37E-08	5.00	0.22E-09	5.00	0.24E-09	5.00
0.0039	0.12E-09	4.99	0.68E-11	5.00	0.74E-11	5.00

P^4 -elements, SSP-RK5 scheme at $T = 1.5$.

Elliptic Equation

Consider a second-order elliptic model problem:

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= u_D \quad \text{on } \partial\Omega. \end{aligned}$$

Introduce an auxiliary variable q and rewrite the equation as

$$\begin{aligned} q &= \nabla u \quad \text{in } \Omega, \\ -\nabla \cdot q &= f \quad \text{in } \Omega, \\ u &= u_D \quad \text{on } \partial\Omega. \end{aligned}$$

Definition

For each element K of the mesh T_h of the domain Ω , we define where \mathbf{n}_K is the outward unit normal to K ,

$$(u, v)_K = \int_K uv \, dx, \quad (1)$$

$$\langle w, v \rangle_{\partial K} = \int_{\partial K} wv \, ds. \quad (2)$$

Weak form and DG method

The approximate solution (q_h, u_h) on the element K is taken in the space $V(K) \times W(K)$ and is defined as the solution, for all $(v, w) \in V(K) \times W(K)$, of the equations

$$\begin{aligned} (q_h, v)_K + (u_h, \nabla \cdot v)_K - \langle \hat{u}_h, \mathbf{n}_K \cdot v \rangle_{\partial K} &= 0, \\ (q_h, \nabla w)_K - \langle \hat{q}_h \cdot \mathbf{n}_K, w \rangle_{\partial K} &= (f, w)_K, \end{aligned}$$

with Dirichlet boundary condition

$$\hat{u}_h = u_D \quad \text{on } \partial K \cap \partial \Omega$$

All the DG methods are generated by choosing the local spaces $V(K) \times W(K)$ and the numerical traces $\hat{q}_h \cdot \mathbf{n}_K$ and \hat{u}_h .

Numerical traces

The definition of the numerical traces \hat{q}_h and \hat{u}_h strongly influences the properties of the corresponding DG method. In this context, we also require that the numerical traces be linear functions of the traces of $q_h \cdot \mathbf{n}_K$ and u_h which are consistent and single valued.

Example (Cockburn and Shu 1998)

$$\begin{aligned}\hat{q}_h &:= \{q_h\} - C_{qq} \llbracket q_h \rrbracket - C_{qu} \llbracket u_h \rrbracket, \\ \hat{u}_h &:= \{u_h\} - C_{uu} \llbracket u_h \rrbracket - \cancel{C_{uq} \llbracket q_h \rrbracket}.\end{aligned}$$

Let us consider DG methods having a numerical trace \hat{u}_h independent of q_h . This allows for the easy elimination of the variable q_h , which can now be expressed in terms of u_h in an elementwise manner, and results in the so-called primal formulation of the method.

Mixed Form of LDG Methods

Find $(\mathbf{q}_h, u_h) \in \mathcal{M}_h \times \mathcal{V}_h$ such that

$$\begin{aligned} a_h(\mathbf{q}_h, \mathbf{v}) + b_h(u_h, \mathbf{v}) &= G_h(\mathbf{v}), \\ -b_h(w, \mathbf{q}_h) + c_h(u_h, w) &= F_h(w), \end{aligned}$$

- The corresponding linear system has the form

$$\begin{bmatrix} A & B \\ -B^t & C \end{bmatrix} \begin{bmatrix} Q \\ U \end{bmatrix} = \begin{bmatrix} G \\ F \end{bmatrix}$$

- A and C are symmetric
- B is antisymmetric.

Numerical Result: Elliptic Interface Problem

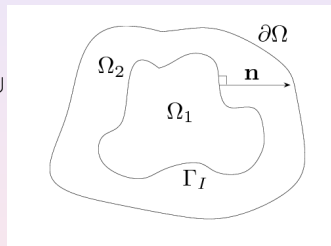
● Model Problem

$$-\nabla \cdot \beta \nabla u = f \text{ in } \Omega_1 \cup \Omega_2$$

$$u = g \text{ on } \partial\Omega,$$

$$u|_{\Omega_1} - u|_{\Omega_2} = a \text{ on } \Gamma_I,$$

$$((\beta \nabla u)|_{\Omega_1} - (\beta \nabla u)|_{\Omega_2}) \cdot \mathbf{n}_e = b \text{ on } \Gamma_I,$$



where f , g , a , and b are functions of x and y , \mathbf{n}_e is the outward unit normal vector to $\partial\Omega_1$, and β is a positive finite constant function on Ω_1 and Ω_2 , separately.

(Chen, Wu, TJM2016)

The Model Problem and LDG method

We assume that $\Omega = [-1, 1] \times [0, 3]$ is the entire domain and Ω_1 is the open interior domain embedded in Ω with a complicated interface

$$\Gamma_I(\theta) = \begin{pmatrix} 0.6 \cos \theta - 0.3 \cos 3\theta \\ 1.5 + 0.7 \sin \theta - 0.07 \sin 3\theta + 0.2 \sin 7\theta \end{pmatrix}$$

for $\theta \in [0, 2\pi]$. The exterior domain is $\Omega_2 = \Omega \setminus \Omega_1$.

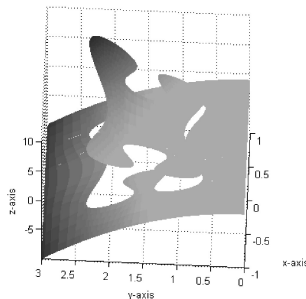
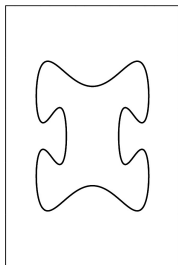


Figure: Domain(left) and numerical solution (right) of test problem 2.

The Model Problem and LDG method

Table: p -convergence of u and q (25 elements) for test problem 2.

Basis function	DoF	L^2 -error of u	L^2 -error of q
Q^1	300	$6.615e - 01$	$2.641e + 01$
Q^2	675	$1.599e - 01$	$1.129e + 01$
Q^3	1200	$4.450e - 02$	$4.346e + 00$
Q^4	1875	$6.711e - 03$	$9.273e - 01$
Q^5	2700	$2.652e - 03$	$4.010e - 01$
Q^6	3675	$3.512e - 04$	$7.475e - 02$
Q^7	4800	$1.095e - 04$	$2.204e - 02$
Q^8	6075	$2.148e - 05$	$5.815e - 03$
Q^9	7500	$3.297e - 06$	$7.967e - 04$
Q^{10}	9075	$9.766e - 07$	$3.258e - 04$
Q^{11}	10800	$8.334e - 08$	$2.376e - 05$

p -convergence and Domain

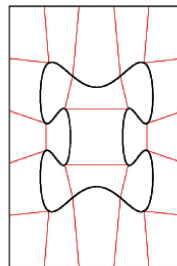
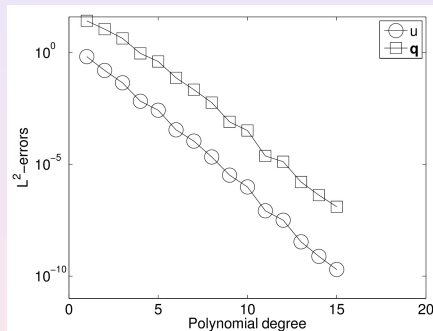


Figure: p -convergence of u and q on a 25-element mesh (left). Corresponding curved-edge quadrilateral mesh with 25 elements (right).

Fenics Project

- The FEniCS Project is a collection of free and open-source software components with the common goal to enable automated solution of differential equations.³
- The components provide scientific computing tools for working with computational meshes, finite-element variational formulations of ordinary and partial differential equations, and numerical linear algebra.
- FEniCS enables users to quickly translate scientific models into efficient finite element code. With the high-level Python and C++ interfaces to FEniCS, it is easy to get started.⁴
- FEniCS runs on a multitude of platforms.

³[https://en.wikipedia.org/wiki/FEniCS Project](https://en.wikipedia.org/wiki/FEniCS_Project)

⁴<https://fenicsproject.org/>

Reference:

- Solving PDEs in Python: The FEniCS Tutorial I (Free)
- Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book (Free PDF)
- Introduction to Automated Modeling with FEniCS by Ridgway Scott
- FENICS Examples (John Burkardt)

Poisson problem (CG): $\int_{\Omega} \nabla u \cdot \nabla v \, dx = \int_{\Omega} f \, v \, dx$

```
%matplotlib inline
from fenics import *
mesh = UnitSquareMesh (8, 8)
V = FunctionSpace (mesh, "Lagrange", 1)
u0 = Expression ("1+x[0]*x[0]+2*x[1]*x[1]", degree =2)
bc = DirichletBC (V, u0, "on_boundary_")
f = Constant (-6.0)
u = TrialFunction (V)
v = TestFunction (V)
a = inner ( grad (u), grad (v))*dx
L = f*v*dx
u = Function (V)
solve (a == L, u, bc)
plot (u)
```

Google Colaboratory (Colab)

What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in your browser, with

- Zero configuration required
- Free access to GPUs
- Easy sharing

Some sample code:

- MyPoissonCG link: CG method for Poisson problem.
- MyPoissonDG link: DG method for Poisson problem.

Reference: <https://github.com/leodenale/FenicsOnColab>

Motivation for HDG methods

However, the DG methods (for second-order elliptic equations) have been criticized because:

- For the same mesh and the same polynomial degree, the number of globally coupled degrees of freedom of the DG methods is much bigger than those of the CG method. Moreover, the orders of convergence of both the vector and scalar variables are also the same.

Elliptic Equation

Consider a second-order elliptic model problem:

$$\begin{aligned} cq + \nabla u &= 0 && \text{in } \Omega, \\ \nabla \cdot q &= f && \text{in } \Omega, \\ \hat{u} &= u_D && \text{on } \partial\Omega. \end{aligned}$$

Here c is a matrix-valued function which is symmetric and uniformly positive definite on Ω

The HDG methods⁵.

Rewrite the equations. If for each $K \in \Omega_h$, we assume that we know the trace \hat{u} on ∂K , we can obtain (q, u) inside K as the solution of
Consider a second-order elliptic model problem:

$$\begin{aligned} cq + \nabla u &= 0 && \text{in } K, \\ \nabla \cdot q &= f && \text{in } K, \\ u &= \hat{u} && \text{on } \partial K. \end{aligned}$$

Then \hat{u} can be determined as the solution, on each edge $F \in \mathcal{E}_h$, of

$$\begin{aligned} \llbracket \hat{q} \cdot n \rrbracket &= 0 && \text{if } F \in \mathcal{E}_h^o, \text{ (interior edges)} \\ \hat{u} &= u_D && \text{if } F \in \mathcal{E}_h^\partial, \text{ (exterior edges)} \end{aligned}$$

Note: $\llbracket \hat{q} \cdot n \rrbracket := \hat{q}^+ \cdot n^+ + \hat{q}^- \cdot n^-$.

⁵*The Hybridizable Discontinuous Galerkin Methods*, Proceedings of the International Congress of Mathematicians 2010, pp.2749-2775.

The local problems: A weak formulation on each element.

On each element $K \in \Omega_h$, we define $(q_h, u_h) \in V(K) \times W(K)$ in terms of (\hat{u}_h, f) such that

$$\begin{aligned}(cq_h, v)_K - (u_h, \nabla \cdot v)_K + \langle \hat{u}_h, v \cdot n \rangle_{\partial K} &= 0, \\ -(q_h, \nabla w)_K + \langle \hat{q}_h \cdot n, w \rangle_{\partial K} &= (f, w)_K,\end{aligned}$$

for all $(v, w) \in V(K) \times W(K)$, where

$$\hat{q}_h \cdot n = q_h \cdot n + \tau(u_h - \hat{u}_h) \quad \text{on } \partial K.$$

Note: $(u, w)_K := \int_K uw \, dx$, $\langle w, v \rangle_{\partial K} = \int_{\partial K} wv \, ds$

The global problem: The weak formulation for \hat{u}_h .

For each face $F \in \mathcal{E}_h^o$, we take $\hat{u}_h|_F$ in the space $M(F)$. We determine \hat{u}_h by requiring that,

$$\begin{aligned} \langle \nu, [\![\hat{q}_h]\!] \rangle_F &= 0, \forall \nu \in M(F) \quad \text{if } F \in \mathcal{E}_h^o, \\ \hat{u}_h &= u_D \quad \text{if } F \in \mathcal{E}_h^\partial. \end{aligned}$$

All the HDG methods are generated by choosing the local spaces $V(K)$, $W(K)$, $M(F)$ and the stabilization function τ .

- Notice that \hat{u}_h is the data of the local problems but is the unknown of the global problem. So, the only globally-coupled degrees of freedom are those of \hat{u}_h .
- By solving the local problems, we express q_h , u_h and \hat{q}_h in terms of \hat{u}_h and f . With these expressions, we construct the matrix equation associated to the global problem.
- After solving it, we can insert the actual values of \hat{u}_h in the expressions we had obtained for q_h , u_h and \hat{q}_h . Next, we describe this procedure more precisely.

The main features of the HDG methods.

- The HDG methods are obtained by discretizing characterizations of the exact solution written in terms of many local problems, one for each element of the mesh Ω_h , with suitably chosen data, and in terms of a single global problem that actually determines them.
- This permits an efficiently implementation since they inherit the above-mentioned structure of the exact solution. This is what renders them efficiently implementable, especially within the framework of hp-adaptive methods, as is typical of DG methods.

HDG3D

- Matlab implementation of the Hybridizable Discontinuous Galerkin method on general tetrahedrizations of polyhedra in three dimensional space.
- Developed by group Team Pancho at the Department of Mathematical Sciences at the University of Delaware.
- Project website: <https://team-pancho.github.io/HDG3D/>

Remark:

- Move “A_simple_example.m” up one level.
- Move the directory “meshes” up one level.

Summary and Reference

Main features of the DG methods:

- High-order accurate.
- Locally Conservative.
- Adaptivity
- High parallelizability,
- The HDG methods are obtained by constructing discrete versions (based on discontinuous Galerkin methods) of the above characterization of the exact solution.
- In this way, the globally coupled degrees of freedom will be those of the corresponding global formulations.

Reference

- B. Cockburn, G. Karniadakis, C.-W. Shu, The development of Discontinuous Galerkin methods, in Discontinuous Galerkin methods. Theory, computation and applications, Lecture Notes in Computational Science and Engineering, Volume 11, Springer, 2000.
- B. Cockburn and C.-W. Shu, Runge-Kutta Discontinuous Galerkin methods for convection-dominated problems, J. Sci. Comput. 16 (2001), pp. 173-261.
- D. Arnold, F. Brezzi, B. Cockburn and D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SINUM 39 (2002), pp. 1749-1779.
- B. Cockburn, Discontinuous Galerkin methods, ZAMM Z. Angew. Math. Mech. 83 (2003), pp. 731-754.
- B. Cockburn, Discontinuous Galerkin methods for computational fluid dynamics, Encyclopedia of Computational Mechanics Second Edition (2018): 1-63.